Discriminative Models for Speech Recognition

M.J.F. Gales
Cambridge University Engineering Department
Trumpington Street,
Cambridge, CB2 1PZ, UK
Email: mjfg@eng.cam.ac.uk

***Abstract*— The vast majority of automatic speech recognition systems use Hidden Markov Models (HMMs) as the underlying acoustic model. Initially these models were trained based on the maximum likelihood criterion. Significant performance gains have been obtained by using discriminative training criteria, such as maximum mutual information and minimum phone error. However, the underlying acoustic model is still generative, with the associated constraints on the state and transition probability distributions, and classification is based on Bayes' decision rule. Recently, there has been interest in examining discriminative, or direct, models for speech recognition. This paper briefly reviews the forms of discriminative models that have been investigated. These include maximum entropy Markov models, hidden conditional random fields and conditional augmented models. The relationships between the various models and issues with applying them to large vocabulary continuous speech recognition will be discussed.**

I. Introduction

Automatic speech recognition (ASR), also known as speech to text transcription (STT), is an interesting statistical processing problem. Compared to many machine learning tasks there is a large amount of training data, billions of words of language model training data and billions of frames of acoustic model training data. In addition ASR is a sequence classification problem. Each sentence is parameterised as a sequence of continuous valued frames, normally at a fixed 10 milli-second frame-rate. Over the years there have been a range of techniques developed for speech recognition. This has allowed large vocabulary continuous speech recognition (LVCSR) tasks, such as Broadcast News transcription [1], to be addressed. Though a number of modifications to the acoustic models have been made, for example speaker adaptation [2], adaptive training [3] and semi-tied covariance matrices [4], the underlying model has remained a Hidden Markov Model (HMM) [5].

One of the major developments that has significantly improved the performance of ASR systems is the use of discriminative criteria for training HMMs, rather than using the Maximum Likelihood (ML) criterion. A number of criteria, such as Maximum Mutual Information (MMI) [6], [7] and Minimum Phone Error (MPE) [8], [9], have been used to train the parameters of the HMM[1]. Initially these criteria were applied to small vocabulary speech recognition tasks. A number of techniques were then developed to enable their use for LVCSR tasks. In particular, schemes such as I-smoothing [8] and language model weakening [10] have been developed to improve generalisation and the use of lattices to compactly represent the denominator score [11].

Though large reductions in word error rate (WER) have been obtained on a range of tasks, the performance on LVCSR tasks, and tasks in challenging acoustic conditions, is still not satisfactory for many speech-enabled applications. This has led to interest in discriminative models for speech recognition [12], [13], [14], [15] where the posterior of the word-sequence given the observation is directly modelled. This paper briefly reviews HMMs, discriminative training criteria, and the current forms of discriminative models that have been applied to ASR. In particular, conditional augmented models are described along with how these may be used for LVCSR tasks.

[1]The HTK hidden Markov model toolkit, available at `http://htk.eng.cam.ac.uk/`, supports many of the current state-of-the-art techniques used in ASR.

II. Hidden Markov Models

Hidden Markov Models (HMMs) [5] are the standard acoustic model used in speech recognition. HMMs comprise a discrete latent space, the state sequence, and associated state output distributions. The observations are assumed to be conditionally independent given the state that generated them.

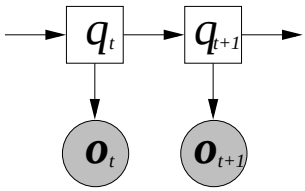

Fig. 1. HMM Dynamic Bayesian Network

Figure 1 shows the dynamic Bayesian network (DBN) associated with an HMM. For simplicity the use of mixtures for each of the states has not been shown. The likelihood of generating the observation sequence $\mathbf{O}_{1:T} = \{\mathbf{o}_1, \ldots, \mathbf{o}_T\}$ with an HMM having model parameter values $\boldsymbol{\lambda}$ is given by

$$p(\mathbf{O}_{1:T}|\mathbf{w};\boldsymbol{\lambda}) = \sum_{\mathbf{q}} P(\mathbf{q}|\mathbf{w}) \prod_{t=1}^{T} p(\mathbf{o}_t|q_t;\boldsymbol{\lambda}^{(\mathbf{w})}) \qquad (1)$$

where $\mathbf{q} = \{q_1, \ldots, q_T\}$ is the state at each time instance and the summation is over all possible state sequences. If mixtures of members of the exponential family, typically Gaussian

distributions, are used, then

$$p(\mathbf{o}_t|q_t;\boldsymbol{\lambda}^{(\mathbf{w})}) = \sum_{m=1}^{M} c_m \mathcal{N}(\mathbf{o}_t;\boldsymbol{\mu}^{(m)},\boldsymbol{\Sigma}^{(m)}) \quad (2)$$

$P(\mathbf{q}|\mathbf{w})$ allows the use of multiple pronunciations and pronunciation probabilities. The state sequences are modelled as a first-order Markov process. The parameters are stored in the transition matrix, $\mathbf{A}$, where $P(q_{t-1}=s_i, q_t=s_j)=a_{ij}$. The standard training of HMM is based on Maximum Likelihood (ML) training. The likelihood criterion may be expressed as

$$\mathcal{F}_{\texttt{ml}}(\boldsymbol{\lambda}) = \frac{1}{R}\sum_{r=1}^{R} \log(p(\mathbf{O}^{(r)}|\mathbf{w}_{\texttt{ref}}^{(r)};\boldsymbol{\lambda})) \quad (3)$$

This optimisation is normally performed using Expectation Maximisation (EM) [16].

During inference, or decoding, classification is based on Bayes' decision rule

$$\hat{\mathbf{w}} = \arg\max_{\mathbf{w}} \{P(\mathbf{w}|\mathbf{O}_{1:T};\boldsymbol{\lambda})\} \quad (4)$$

where the word sequence posterior is obtained using Bayes' rule

$$P(\mathbf{w}|\mathbf{O}_{1:T};\boldsymbol{\lambda}) = \frac{1}{Z(\boldsymbol{\lambda},\mathbf{O}_{1:T})} p(\mathbf{O}_{1:T}|\mathbf{w};\boldsymbol{\lambda}^{(\mathbf{w})})P(\mathbf{w}) \quad (5)$$

$Z(\boldsymbol{\lambda},\mathbf{O}_{1:T})$ is the normalisation term. As $Z(\boldsymbol{\lambda},\mathbf{O}_{1:T})$ is independent of the hypothesised word sequence, it is normally ignored. Viterbi decoding is often used for this process [5].

III. Discriminative Training Criteria

For ML to be the "best" training criterion, the data and models are assumed to satisfy a number of requirements, for example the quantity of training data available and model-correctness [17]. These requirements are not satisfied when modelling speech data. This has led to the use of discriminative training criteria, which are more closely linked to minimising the error rate, rather than maximising the likelihood of generating the training data. For speech recognition three main forms of discriminative training have been examined. Note for speech recognition the language model (or class prior), $P(\mathbf{w})$, is not normally trained in conjunction with the acoustic model (though there has been some work in this area [18]). Typically the amount of text training data for the language model is far greater (orders of magnitude) than the available acoustic training data.

Maximum Mutual Information (MMI): the following form [6], [7] is maximised

$$\mathcal{F}_{\texttt{mmi}}(\boldsymbol{\lambda}) = \frac{1}{R}\sum_{r=1}^{R} \log(P(\mathbf{w}_{\texttt{ref}}^{(r)}|\mathbf{O}^{(r)};\boldsymbol{\lambda})) \quad (6)$$

where $\mathbf{O}^{(r)}$ is the r^{th} training utterance with transcription $\mathbf{w}_{\texttt{ref}}^{(r)}$. This equates to maximising the mutual information between the observed sequences and the models[2].

[2]Given that the class priors are fixed this should really be called conditional entropy training. When this form of training criterion is used with discriminative models it is also known as Conditional Maximum Likelihood (CML) training.

Minimum Classification Error (MCE): is a smooth measure of the error [19]. This is normally based on a smooth function of the difference between the log-likelihood of the correct sequence and all other competing word sequences. This may be expressed in terms of the posteriors as

$$\mathcal{F}_{\texttt{mce}}(\boldsymbol{\lambda}) = \frac{1}{R}\sum_{r=1}^{R}\left(\frac{1}{1+\left[\frac{P(\mathbf{w}_{\texttt{ref}}^{(r)}|\mathbf{O}^{(r)};\boldsymbol{\lambda})}{\sum_{\mathbf{w}\neq\mathbf{w}_{\texttt{ref}}^{(r)}} P(\mathbf{w}|\mathbf{O}^{(r)};\boldsymbol{\lambda})}\right]^{\varrho}}\right) \quad (7)$$

There are some important differences between MCE and MMI. The first is that the denominator term does not include the correct word sequence. Second the posteriors (or log-likelihoods) are smoothed with a sigmoid function, which introduces an additional smoothing term ϱ. When $\varrho=1$ then

$$\mathcal{F}_{\texttt{mce}}(\boldsymbol{\lambda}) = 1 - \frac{1}{R}\sum_{r=1}^{R} P(\mathbf{w}_{\texttt{ref}}^{(r)}|\mathbf{O}^{(r)};\boldsymbol{\lambda}) \quad (8)$$

Minimum Bayes' Risk (MBR): rather than trying to model the correct distribution, as in the MMI criterion, the expected loss during inference is minimised [20], [21]. Here

$$\mathcal{F}_{\texttt{mbr}}(\boldsymbol{\lambda}) = \frac{1}{R}\sum_{r=1}^{R}\sum_{\mathbf{w}} P(\mathbf{w}|\mathbf{O}^{(r)};\boldsymbol{\lambda})\mathcal{L}(\mathbf{w},\mathbf{w}_{\texttt{ref}}^{(r)}) \quad (9)$$

where $\mathcal{L}(\mathbf{w},\mathbf{w}_{\texttt{ref}}^{(r)})$ is the loss function of word sequence $\mathbf{w}$ against the reference for sequence r, $\mathbf{w}_{\texttt{ref}}^{(r)}$. There are a number of loss functions that have been examined.

- **1/0 loss**: for continuous speech recognition this is equivalent to a sentence-level loss function.

$$\mathcal{L}(\mathbf{w},\mathbf{w}_{\texttt{ref}}^{(r)}) = \begin{cases} 1; & \mathbf{w}\neq\mathbf{w}_{\texttt{ref}}^{(r)} \\ 0; & \mathbf{w}=\mathbf{w}_{\texttt{ref}}^{(r)} \end{cases}$$

  When $\varrho=1$ MCE and MBR training with a sentence cost function are the same.
- **Word**: the loss function directly related to minimising the expected Word Error Rate (WER). It is normally computed by minimising the Levenshtein edit distance.
- **Phone**: for large vocabulary speech recognition not all word sequences will be observed. To help the generalisation the loss function is often computed between the phone sequences, rather than word sequences. In the literature this is known as Minimum Phone Error (MPE) training [8], [9].

It is also possible to base the loss function on the specific task for which the classifier is being built [21].

A comparison of the MMI, MCE and MBR criteria on the Wall Street Journal (WSJ) task and a general framework is given in [22]. Both MCE and MPE were found to outperform MMI on this task. To enable these discriminative training criteria to be successfully applied to LVCSR tasks, a number of techniques have been developed to improve generalisation. These include:

Acoustic de-weighting: for all forms of discriminative criteria described, the log-likelihoods are often scaled. This is because the dynamic range of the likelihoods obtained from the HMMs are typically far greater than they should be, due to the conditional independence assumptions. Furthermore the language model may also be scaled due to is mismatch. For clarity these scaling factors have not been included in the discriminative training formulae given.

Language model simplification: the form of the language model used in training should in theory match the form used for inference. However, it has been found that using simpler models, unigrams or heavily pruned bigrams, for training despite using trigrams or fourgrams in decoding improves performance [10]. By weakening the language model, the number of possible confusions is increased allowing more complex models to be trained given a fixed quantity of training data.

I-smoothing: to improve the generalisation "robust" parameter priors may be used when estimating the models. These priors may either be based on the ML parameter estimates [8] or, for example when using MPE training, the MMI estimates [23]. For MPE this was found to essential to achieve performance gains [8].

In all the criteria described the optimisation criterion is a function of the word sequence posterior. Thus the criteria have some of the attributes of the direct or discriminative models. However the underlying acoustic model itself is still a generative model, with word sequence posteriors being produced using Bayes' rule. Note in recent years MBR decoding, associated normally with the word-level cost function, has become popular in speech recognition [24], [25], [26].

IV. Large Margin HMMs

Recently there has been interest in using large margin training approaches [27], [28], [29], [30]. The simplest form of large margin training criterion can be expressed as maximising[3]

$$\mathcal{F}_{\texttt{mm}}(\boldsymbol{\lambda}) = \frac{1}{R}\sum_{r=1}^{R}\left(\min_{\mathbf{w}\neq\mathbf{w}_{\texttt{ref}}^{(r)}}\left\{\log\left(\frac{P(\mathbf{w}_{\texttt{ref}}^{(r)}|\mathbf{O}^{(r)};\boldsymbol{\lambda})}{P(\mathbf{w}|\mathbf{O}^{(r)};\boldsymbol{\lambda})}\right)\right\}\right) \quad (10)$$

This aims to maximise the minimum distance between the log-posterior of the correct label and all incorrect labels[4]. This criterion has properties related to both the MMI and MCE criterion. A log-posterior cost function is used, as in the MMI criterion, rather than the posterior-based MCE and MBR criteria. However, the denominator term used with this large margin approach does not include an element from the correct label in a similar fashion to the MCE criterion in equation 7.

A couple of variants of large margin training have been proposed. For binary classification tasks a modified version of SVM training has been used to train the kernel parameters [27]. When the kernel is constrained to be a simple log-likelihood score (rather than using derivatives as discussed in section VII) this is the same as training a maximum margin HMM for a binary task. In [29] a minimum margin size constraint was introduced for the multi-class problem.. Here a "hinge-loss" function of the following form is minimised[5]

$$\mathcal{F}_{\texttt{hl}}(\boldsymbol{\lambda}) = \frac{1}{R}\sum_{r=1}^{R}\left[\rho - \min_{\mathbf{w}\neq\mathbf{w}_{\texttt{ref}}^{(r)}}\left\{\log\left(\frac{P(\mathbf{w}_{\texttt{ref}}^{(r)}|\mathbf{O}^{(r)};\boldsymbol{\lambda})}{P(\mathbf{w}|\mathbf{O}^{(r)};\boldsymbol{\lambda})}\right)\right\}\right]_{+} \quad (11)$$

where

$$[f(x)]_{+} = \begin{cases} f(x) & \text{if } f(x) > 0 \\ 0 & \text{otherwise} \end{cases} \quad (12)$$

and ρ determines the size of the margin. The use of an approximate word level criterion, which out-performs the utterance level, is also discussed in the paper.

Alternatively in [30], the size of the margin is specified in terms of a loss function between the two sets of sequences. Here

$$\mathcal{F}_{\texttt{ha}}(\boldsymbol{\lambda}) = \frac{1}{R}\sum_{r=1}^{R}\left[\max_{\mathbf{w}\neq\mathbf{w}_{\texttt{ref}}^{(r)}}\left\{\mathcal{H}(\mathbf{w},\mathbf{w}_{\texttt{ref}}^{(r)}) - \log\left(\frac{P(\mathbf{w}_{\texttt{ref}}^{(r)}|\mathbf{O}^{(r)};\boldsymbol{\lambda})}{P(\mathbf{w}|\mathbf{O}^{(r)};\boldsymbol{\lambda})}\right)\right\}\right]_{+} \quad (13)$$

where $\mathcal{H}(\mathbf{w},\mathbf{w}_{\texttt{ref}}^{(r)})$ is the Hamming distance, measured at the frame, between the two sequences. Note, this is the Minimum Phone Frame Error (MPFE) loss function used with minimum Bayes' error training in [33] where it was shown to yield small, but consistent, gains over MPE. To simplify the optimisation problem, the minimum in equation 13 is replaced by a summation. This yields an upper bound, which is minimised, of the form

$$\mathcal{F}_{\texttt{ha}}(\boldsymbol{\lambda}) \leq \frac{1}{R}\sum_{r=1}^{R}\left[-\log\left(P(\mathbf{w}_{\texttt{ref}}^{(r)}|\mathbf{O}^{(r)};\boldsymbol{\lambda})\right) + \log\left(\sum_{\mathbf{w}}P(\mathbf{w}|\mathbf{O}^{(r)};\boldsymbol{\lambda})\mathcal{L}(\mathbf{w},\mathbf{w}_{\texttt{ref}}^{(r)})\right)\right]_{+} \quad (14)$$

where the loss function is defined as

$$\mathcal{L}(\mathbf{w},\mathbf{w}_{\texttt{ref}}^{(r)}) = \begin{cases} \exp(\mathcal{H}(\mathbf{w},\mathbf{w}_{\texttt{ref}}^{(r)})); & \mathbf{w}\neq\mathbf{w}_{\texttt{ref}}^{(r)} \\ 0; & \mathbf{w}=\mathbf{w}_{\texttt{ref}}^{(r)} \end{cases}$$

It is interesting to compare the criterion in equation 14 with the discriminative criteria in the previous section. The first term within the hinge function is the negated log-posterior, the same as the MMI-criterion. The second term is a log'd

[3]Slightly modified versions of the large margin criteria given in the references are usedin this paper. This allows the inclusion of language model score and makes contrasts with the other discriminative criteria easier. If the language model scores are ignored the forms given in the refernces will be obtained. Note for equation 10 it is also possible to use the posterior, rather than the log-posterior, for the margin. It is worth noting that a similar criterion was discussed in [31].

[4]For a discussion of maximum margin training with discrete sequence data see [32].

[5]In [28] a simple form of hinge-like loss is used by only selecting a subset of the training data to estimate the model.

version of the MBR-criterion, but with a modified version of the MPFE loss function. Furthermore this is passed through a hinge-loss function. In [30], [34] further approximations to the likelihood are made which yield a convex optimisation problem. However, as part of this formulation the constraint that the resultant model is valid generative model is lost. Thus with the additional approximations a discriminative model is trained with features having the same dependencies as the HCRF in section VI, rather than using a discriminative criterion to train an HMM.

To date have been few attempts to apply these large margin criteria to large vocabulary speech recognition tasks. In [29] there are initial experiments on the WSJ task in which a reduction in WER were obtained compared to an ML baseline system. However there are no contrasts with other discriminative approaches and the baseline system is not a state-of-the-art system.

V. Maximum Entropy Markov Models

The DBN in figure 1 may be modified to produce a discriminative (or direct model) by reversing the direction of the arcs from the states to the observations and using an exponential model. This is known as a Maximum Entropy Markov Model (MEMM) [12].

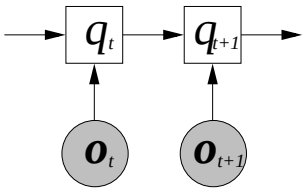

Fig. 2. MEMM Dynamic Bayesian Network

Figure 2 shows the DBN for this form of model. The posterior associated with word sequence $\mathbf{w}$ is given by

$$P(\mathbf{w}|\mathbf{O}_{1:T};\boldsymbol{\alpha}) = \sum_{\mathbf{q}} P(\mathbf{w}|\mathbf{q}) \prod_{t=1}^{T} P(q_t|\mathbf{o}_t, q_{t-1};\boldsymbol{\alpha}) \quad (15)$$

An exponential model is used for the state posterior distribution

$$P(q_t|\mathbf{o}_t, q_{t-1};\boldsymbol{\alpha}) = \frac{1}{Z(\boldsymbol{\alpha}, \mathbf{o}_t)} \exp\left(\boldsymbol{\alpha}^{\mathsf{T}} \mathbf{T}(\mathbf{o}_t, q_t, q_{t-1})\right) \quad (16)$$

$P(\mathbf{w}|\mathbf{q})$ is of interest when homophones are present in the language. It can be simply derived from the pronunciation probabilities and Bayes' rule.

One of the issues with this form of model is that there is no elegant approach to incorporating a language model in this framework. This has limited possible gains with this form of model [12].

VI. Hidden Conditional Random Field

Conditional Random Fields (CRFs) [35] are one approach to constructing discriminative models. Given the observation sequence $\mathbf{O}_{1:T} = \{\mathbf{o}_1, \ldots, \mathbf{o}_T\}$ and label sequence $\mathbf{w} = \{w_1, \ldots, w_L\}$, the standard form for this model is

$$P(\mathbf{w}|\mathbf{O}_{1:T};\boldsymbol{\alpha}) = \frac{1}{Z(\boldsymbol{\alpha}, \mathbf{O}_{1:T})} \exp\left(\boldsymbol{\alpha}^{\mathsf{T}} \mathbf{T}(\mathbf{O}_{1:T}, \mathbf{w})\right) \quad (17)$$

where $Z(\boldsymbol{\alpha}, \mathbf{O}_{1:T})$ is the appropriate normalisation term to ensure a valid PMF. For some applications $L = T$ and it is possible to extract the standard transition and state features. However for tasks such as speech recognition often $T > L$ since the sample rate of the observations is fixed. This has led to the use of Hidden CRFs (HCRFs) for speech recognition [13].

The general probability form when introducing a latent variable to the the CRF framework is[6]

$$P(\mathbf{w}|\mathbf{O}_{1:T};\boldsymbol{\alpha}) = \frac{1}{Z(\boldsymbol{\alpha}, \mathbf{O}_{1:T})} \sum_{\mathbf{q}} \exp\left(\boldsymbol{\alpha}^{\mathsf{T}} \mathbf{T}(\mathbf{O}_{1:T}, \mathbf{w}, \mathbf{q})\right) \quad (18)$$

where again the summation is over all possible state sequences. The problem is to extract the "appropriate" statistics, $\mathbf{T}(\mathbf{O}_{1:T}, \mathbf{w}, \mathbf{q})$ for recognition. These may be split into two blocks

$$\mathbf{T}(\mathbf{O}_{1:T}, \mathbf{w}, \mathbf{q}) = \begin{bmatrix} \mathbf{T}_{\mathtt{l}}(\mathbf{w}) \\ \mathbf{T}_{\mathtt{a}}(\mathbf{O}_{1:T}, \mathbf{w}, \mathbf{q}) \end{bmatrix} \quad (19)$$

The first set of statistics are associated with the "language model" for this data. These could be estimated in a discriminative fashion for some tasks [18], however the simplest approach is to set

$$\mathbf{T}_{\mathtt{l}}(\mathbf{w}) = \log\left(P(\mathbf{w})\right) \quad (20)$$

and constrain all the α's associated with this for all models to be the same, $\alpha_{\mathtt{l}}$.

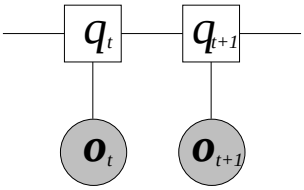

Fig. 3. Feature Dependencies in the HCRF

The second set of statistics are those associated with the acoustic data. There are a number of possible statistics that could be extracted from the sequences. In HCRFs this selection is simplified by using the features that result from the dependencies shown in figure 3. The form of statistics used with the current applications of HCRF to speech recognition

[6]Here the previously mentioned multiple pronunciation and homophone issues have been ignored for clarity. Thus there is a unique one-to-one mapping from the word sequence to the state sequence.

are thus

$$\mathbf{T}_{\mathtt{a}}(\mathbf{O}_{1:T}, \mathbf{w}, \mathbf{q}) = \begin{bmatrix} \vdots \\ \sum_{t=1}^{T} \delta(q_{t-1} - s_i)\delta(q_t - s_i) \\ \sum_{t=1}^{T} \delta(q_t - s_i) \\ \sum_{t=1}^{T} \delta(q_t - s_i)\mathbf{o}_t \\ \sum_{t=1}^{T} \delta(q_t - s_i)\mathrm{vec}(\mathbf{o}_t\mathbf{o}_t^{\mathsf{T}}) \\ \vdots \end{bmatrix} \quad (21)$$

for all i, j. Though these are the same statistics as for a standard HMM, there are now no constraints that the individual state distributions associated with these statistics are valid PDFs. This additional flexibility was found to yield large gains on the TIMIT phone classification task [13] [7].

VII. Dynamic Kernels

The maximum entropy Markov model and the hidden CRF model have the same conditional independence assumptions as the HMM. They differ only in the form of the arcs in the DBN rather than altering the general structure. This means that the statistics extracted from the observation sequence are based on the observation at time t, $\boldsymbol{o}_t$ and the current and previous states. There are a number of approaches to extending the range of dependencies. The easiest approach is to hypothesise possible dependencies and then select the dependencies that improve discrimination most. This is the approach adopted in Buried Markov Models [36]. One interesting aspect of handling speech data is that, since sequences of observations are being classified, the space of possible dependencies is very large making the choice of an appropriate hypothesised subset hard. An alternative approach is to use some of the approaches adopted with dynamic kernels to give a systematic way of extracting features from the sequences.

A number of kernels have been proposed for handling sequence data, including marginalised count kernels [37], Fisher kernels [38], string kernels [39] and generative kernels [27]. An interesting class of these sequence kernels are based on generative models. Both Fisher kernels [38] and generative kernels [27] fall in this category, and make use of generative models to map the variable length sequences to a fixed dimensionality. In generative kernels the feature space used has the form

$$\boldsymbol{\phi}(\mathbf{O}_{1:T}; \boldsymbol{\lambda}) = \begin{bmatrix} \log(p(\mathbf{O}_{1:T}; \boldsymbol{\lambda})) \\ \boldsymbol{\nabla}_{\lambda} \log(p(\mathbf{O}_{1:T}; \boldsymbol{\lambda})) \\ \vdots \\ \boldsymbol{\nabla}_{\lambda}^{\rho} \log(p(\mathbf{O}_{1:T}; \boldsymbol{\lambda})) \end{bmatrix} \quad (22)$$

where ρ is the order of the kernel, $\boldsymbol{\lambda}$ specifies the parameters of the generative model.

It is useful to examine the form of the derivatives when a discrete HMM is used as the generative model. Initially consider the differential with respect to bin m of state s_j (equivalent in the continuous case to the prior of component m in state s_j). The first derivative is given by

$$\boldsymbol{\nabla}_{c_{jm}} \log(p(\mathbf{O}_{1:T}; \boldsymbol{\lambda})) = \sum_{t=1}^{T} (\gamma_{jm}(t)/c_{jm} - \gamma_j(t)) \quad (23)$$

and

$$\gamma_{jm}(t) = P(q_t = s_{jm} | \mathbf{O}_{1:T}; \boldsymbol{\lambda}) \quad (24)$$

$$\gamma_j(t) = \sum_{m=1}^{M} \gamma_{jm}(t) \quad (25)$$

For the second derivative,

$$\boldsymbol{\nabla}_{c_{kn}} \boldsymbol{\nabla}_{c_{jm}}^{\mathsf{T}} \log(p(\mathbf{O}_{1:T}; \boldsymbol{\lambda})) = \quad (26)$$
$$\frac{1}{c_{jm}c_{kn}} \sum_{t=1}^{T} \sum_{\tau=1}^{T} \Big(D(q_t^{(jm)}, q_\tau^{(kn)}) - c_{jm} D(q_t^{(j)}, q_t^{(jm)}) - c_{kn} D(q_t^{(jm)}, q_\tau^{(k)}) + c_{jm}c_{kn} D(q_t^{(j)}, q_\tau^{(k)}) \Big)$$
$$- \frac{2}{c_{jm}c_{kn}} \sum_{t=1}^{T} P(q_t = s_{jm} | \mathbf{O}_{1:T}; \boldsymbol{\lambda}) \delta(s_{jm} - s_{kn})$$

where

$$D(q_t^{(jm)}, q_\tau^{(kn)}) = \quad (27)$$
$$P(q_t = s_{jm}, q_\tau = s_{kn} | \mathbf{O}_{1:T}; \boldsymbol{\lambda}) - \gamma_{jm}(t)\gamma_{kn}(\tau)$$

These features are functions of the complete observation sequence as they depend on the state posterior $\gamma_{jm}(t)$ which is a function of $\mathbf{O}_{1:T}$. Thus long-term dependencies may be represented by these forms of features.

To illustrate the advantage of using the higher order derivatives consider the simple example of a two class problem with a two discrete output symbols $\{\mathtt{A}, \mathtt{B}\}$. The training examples (equally distributed) are:

- Class ω_1: AAAA, BBBB
- Class ω_2: AABB, BBAA

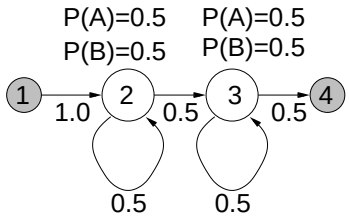

Fig. 4. Example discrete HMM topology and state probabilities

If the discrete two-emitting state HMM shown in figure 4 is trained on this data then the state distributions also shown in figure 4 are obtained. This ML trained HMM is unable to distinguish between the sequences from class ω_1 and ω_2.

Table I shows the values of some of elements of the feature vector associated with a generative kernel for each of the two classes. It is clear that using the first and higher order derivatives of the log-likelihood allow the two classes to be separated, in some cases using a simple linear classifier. From

[7] It should be noted that the training of these models is more complicated than discriminative training of standard HMMs. Gradient decent based optimisation schemes are used which may yield better performance when discriminatively training standard HMMs than the commonly used extended Baum-Welch training.

Feature	Class ω_1		Class ω_2	
	AAAA	BBBB	AABB	BBAA
Log-Lik	-1.11	-1.11	-1.11	-1.11
∇_{2A}	0.50	-0.50	0.33	-0.33
$\nabla_{2A}\nabla'_{2A}$	-3.83	0.17	-3.28	-0.61
$\nabla_{2A}\nabla'_{3A}$	-0.17	-0.17	-0.06	-0.06

TABLE I

FEATURE VECTOR VALUES FOR A GENERATIVE KERNEL

the $\nabla_{2A}\nabla'_{3A}$ row of the table the feature captures the obvious difference between the two classes that the label changes part way through.

One form of discriminative classifier that has been found to yield good empirical results on a range of tasks is the Support Vector Machine (SVM) [40]. By using these generative kernel features SVMs can be applied to binary classification tasks with sequence data. This approach has been applied in the speech processing area to simple small vocabulary speech recognition tasks [15], LVCSR tasks by making use of the acoustic code-breaking framework [41], [42] and speaker verification [43], [44]. The kernel between two sequences, $\mathbf{O}^{(1)}$ and $\mathbf{O}^{(2)}$, has the form

$$K(\mathbf{O}^{(1)}, \mathbf{O}^{(2)}; \boldsymbol{\lambda}) = \boldsymbol{\phi}(\mathbf{O}^{(1)}; \boldsymbol{\lambda})^{\mathsf{T}} \mathbf{G}^{-1} \boldsymbol{\phi}(\mathbf{O}^{(2)}; \boldsymbol{\lambda}) \tag{28}$$

where $\mathbf{G}$ defines the metric. An interesting aspect of these generative kernels is that estimating the decision boundary may be related to estimating the parameters of an Augmented Statistical model [45], [42]. Though good performance has been obtained, it is non-trivial to apply this to tasks with large numbers of classes (without the use of schemes such as acoustic code-breaking).

VIII. CONDITIONAL AUGMENTED MODELS

Conditional Augmented models (CAUG) combine some of the properties of CRF/HCRFs and Dynamic kernels [14]. Rather than restricting the statistics to be the same as those of a standard HMM, generative kernels are used to extract "features" for use in a discriminative classifier. Using the features associated with generative kernels gives an elegant way of combining generative and discriminative models. Now the features associated with the acoustic data are

$$\mathbf{T}_{\mathtt{a}}(\mathbf{O}_{1:T}, \mathbf{w}) = \boldsymbol{\phi}(\mathbf{O}_{1:T}; \boldsymbol{\lambda}) \tag{29}$$

where $\boldsymbol{\lambda}$ are the parameters of the kernel which are a function of the label sequence $\mathbf{w}$. It is now possible to directly use these features in a discriminative exponential model. Consider the simplest form of dynamic kernel, the log-likelihood (zeroth order generative kernel).

$$\mathbf{T}_{\mathtt{a}}(\mathbf{O}_{1:T}, \mathbf{w}) = \begin{bmatrix} \vdots \\ \delta(\mathbf{w} - \tilde{\mathbf{w}}) \log\left(p(\mathbf{O}_{1:T}; \boldsymbol{\lambda}^{(\tilde{\mathbf{w}})})\right) \\ \vdots \end{bmatrix} \tag{30}$$

for all $\tilde{\mathbf{w}}$ This then yields the following form of posterior

$$P(\mathbf{w}|\mathbf{O}_{1:T}; \boldsymbol{\alpha}, \boldsymbol{\lambda}) = \frac{1}{Z(\boldsymbol{\alpha}, \boldsymbol{\lambda}, \mathbf{O}_{1:T})} \times \exp\left(\alpha_1 \log(P(\mathbf{w})) + \alpha^{(\mathbf{w})} \log\left(p(\mathbf{O}_{1:T}; \boldsymbol{\lambda}^{(w)})\right)\right) \tag{31}$$

This is similar to the posterior obtained for a standard HMM, see equation 5, though now the values of $\boldsymbol{\alpha}$ may also be optimised[8].

Using more powerful dynamic kernels yields interesting extensions to the standard HMM. Consider a first order generative kernel.

$$\mathbf{T}_{\mathtt{a}}(\mathbf{O}_{1:T}, \mathbf{w}) = \begin{bmatrix} \vdots \\ \delta(\mathbf{w} - \tilde{\mathbf{w}}) \log(p(\mathbf{O}_{1:T}; \boldsymbol{\lambda}^{(\tilde{\mathbf{w}})})) \\ \vdots \\ \delta(\mathbf{w} - \tilde{\mathbf{w}}) \boldsymbol{\nabla}_{\lambda} \log(p(\mathbf{O}_{1:T}; \boldsymbol{\lambda}^{(\tilde{\mathbf{w}})})) \\ \vdots \end{bmatrix} \tag{32}$$

for all $\tilde{\mathbf{w}}$ The posterior is then expressed as

$$P(\mathbf{w}|\mathbf{O}_{1:T}; \boldsymbol{\alpha}, \boldsymbol{\lambda}) = \frac{1}{Z(\boldsymbol{\alpha}, \boldsymbol{\lambda}, \mathbf{O}_{1:T})} \times \exp\left(\alpha_1 \log(P(\mathbf{w})) + \boldsymbol{\alpha}^{\mathsf{T}} \mathbf{T}_{\mathtt{a}}(\mathbf{O}_{1:T}, \mathbf{w}, \boldsymbol{\lambda})\right) \tag{33}$$

This allows the longer-term dependencies in the derivative parameters to be incorporated.

It is useful at this stage to examine the form of features (or statistics) that are obtained using an HMM with continuous observation feature vectors as the generative model. Consider the form based on a first order kernel.

$$\mathbf{T}_{\mathtt{a}}(\mathbf{O}_{1:T}, \mathbf{w}) = \begin{bmatrix} \vdots \\ \delta(\mathbf{w} - \tilde{\mathbf{w}}) \log(p(\mathbf{O}_{1:T}; \boldsymbol{\lambda}^{(\tilde{\mathbf{w}})})) \\ \delta(\mathbf{w} - \tilde{\mathbf{w}}) \boldsymbol{\nabla}_{\mu_j} \log(p(\mathbf{O}_{1:T}; \boldsymbol{\lambda}^{(\tilde{\mathbf{w}})})) \\ \delta(\mathbf{w} - \tilde{\mathbf{w}}) \boldsymbol{\nabla}_{\Sigma_j} \log(p(\mathbf{O}_{1:T}; \boldsymbol{\lambda}^{(\tilde{\mathbf{w}})})) \\ \delta(\mathbf{w} - \tilde{\mathbf{w}}) \boldsymbol{\nabla}_{a_{ij}} \log(p(\mathbf{O}_{1:T}; \boldsymbol{\lambda}^{(\tilde{\mathbf{w}})})) \\ \vdots \end{bmatrix} \tag{34}$$

for all state pairings i and j where

$$\boldsymbol{\nabla}_{\mu_j} \log(p(\mathbf{O}_{1:T}; \boldsymbol{\lambda})) = \sum_{t=1}^{T} \gamma_j(t) \boldsymbol{\Sigma}_j^{-1} \left(\mathbf{o}_t - \boldsymbol{\mu}_j\right) \tag{35}$$

$$\boldsymbol{\nabla}_{\Sigma_j} \log(p(\mathbf{O}_{1:T}; \boldsymbol{\lambda})) = \frac{1}{2} \sum_{t=1}^{T} \gamma_j(t) \text{vec}\left(-\boldsymbol{\Sigma}_j^{-1} + \boldsymbol{\Sigma}_j^{-1} (\mathbf{o}_t - \boldsymbol{\mu}_j)(\mathbf{o}_t - \boldsymbol{\mu}_j)^{\mathsf{T}} \boldsymbol{\Sigma}_j^{-1}\right) \tag{36}$$

$$\boldsymbol{\nabla}_{a_{ij}} \log(p(\mathbf{O}_{1:T}; \boldsymbol{\lambda})) = \sum_{t=1}^{T} (P(q_{t-1} = s_i, q_t = s_j | \mathbf{O}_{1:T}; \boldsymbol{\lambda}) / a_{ij} - \gamma_i(t-1)) \tag{37}$$

In a similar fashion to the discrete example in section VII, these features are again a function of the complete observation sequence $\mathbf{O}_{1:T}$.

[8] In practice these values are fixed to improve the generalisation of the discriminatively trained models.

The CAUG model has two distinct sets of parameters to estimate, the generative model for obtaining the features, $\boldsymbol{\lambda}$, and the parameters of the discriminative model, $\boldsymbol{\alpha}$. The simultaneous optimisation of both parameters is difficult since, for example, the CML objective function has many local maxima. Alternatively the generative model parameters can be estimated using either ML, or one of the discriminative training criteria. Given the estimated value of $\boldsymbol{\lambda}$ and hence associated features, the estimation of $\boldsymbol{\alpha}$ is a convex optimisation problem.

The form of model in equation 33 can be directly used for isolated speech recognition tasks. However for more complex continuous tasks, a problem with this form of model is that the features, training and inference is a function of all the words in the sequence, $\mathbf{w}$. This means that Viterbi decoding may not be used, as the conditional independence assumptions for its efficient implementation are not present in this form of model. One approach to dealing with this is to add an additional level of latent variables, $\boldsymbol{\theta}$. The extracted features are assumed to be conditionally independent given $\boldsymbol{\theta}$, so

$$P(\mathbf{w}|\mathbf{O}_{1:T};\boldsymbol{\alpha},\boldsymbol{\lambda}) = \frac{1}{Z(\boldsymbol{\alpha},\boldsymbol{\lambda},\mathbf{O}_{1:T})}\times \tag{38}$$
$$\sum_{\theta}\exp\left(\boldsymbol{\alpha}_1^{\mathsf{T}}\left[\begin{array}{c}\log(P(\mathbf{w}))\\ \mathbf{T}_1(\boldsymbol{\theta})\end{array}\right] + \sum_{i=1}^{L}\boldsymbol{\alpha}^{\mathsf{T}}\mathbf{T}_{\mathtt{a}}(\mathbf{O}_{t(\mathbf{w},i,\theta)},w_i,\boldsymbol{\lambda})\right)$$

where $\boldsymbol{\theta}$ segments the observation sequence into the L labels (these may be at the word, phone or state level)

$$\mathbf{O}_{1:T} = \left\{\mathbf{O}_{t(\mathbf{w},1,\theta)},\ldots,\mathbf{O}_{t(\mathbf{w},L,\theta)}\right\} \tag{39}$$

This is similar to the HCRF, but the features that are extracted may span many frames. If the sufficient statistics extracted have the same form as the standard HMM (thus model $\boldsymbol{\lambda}$ is not used), then this is now very similar to the "standard" form of hidden CRF.

Equation 38 is still inefficient for training and inference as all possible va;ues of $\boldsymbol{\theta}$ must be considered. This may be addressed, by selecting the best segmentation using the generative model with parameters $\boldsymbol{\lambda}$

$$\hat{\boldsymbol{\theta}} = \arg\max_{\theta}\left\{P(\boldsymbol{\theta})p(\mathbf{O}_{1:T}|\boldsymbol{\theta};\boldsymbol{\lambda})\right\} \tag{40}$$

Equation 38 can be rewritten as

$$P(\mathbf{w}|\mathbf{O}_{1:T};\boldsymbol{\alpha},\boldsymbol{\lambda}) = \frac{1}{Z(\boldsymbol{\alpha},\boldsymbol{\lambda},\mathbf{O}_{1:T})}\times \tag{41}$$
$$\exp\left(\boldsymbol{\alpha}_1^{\mathsf{T}}\left[\begin{array}{c}\log(P(\mathbf{w}))\\ \mathbf{T}_1(\hat{\boldsymbol{\theta}})\end{array}\right] + \sum_{i=1}^{L}\boldsymbol{\alpha}^{\mathsf{T}}\mathbf{T}_{\mathtt{a}}(\mathbf{O}_{t(\mathbf{w},i,\hat{\theta})},w_i,\boldsymbol{\lambda})\right)$$

Training and inference can now be implemented in a similar fashion to the discriminative training implementation in HTK [46]. Initially a lattice is generated using the current model $\boldsymbol{\lambda}$. This is then "model-marked" where time-stamps are added to the lattice at the model-level, this may be either at the phone or word level. In training statistics are then accumulated given these fixed segment boundaries. In inference the best path is found given these fixed boundaries. This is discussed in more detail in [47].

IX. Preliminary CAUG Experiments

This section presents some preliminary experimental results on the TIMIT classification task taken from [14]. The experimental setup described in [13] was used. Models were trained with three states and either ten or twenty mixture-components. Acoustic model decoding was performed without the use of a language model. No data or feature whitening was performed.

Classifier	Criterion		Components	
	$\boldsymbol{\lambda}$	$\boldsymbol{\alpha}$	10	20
HMM	ML	–	29.4	27.3
CAug	ML	CML	24.2	–
HMM	MMI	–	25.3	24.8
CAug	MMI	CML	23.4	–

TABLE II
CLASSIFICATION ERROR ON THE TIMIT CORE TEST SET

The classification (i.e. known phone boundaries) performance on the TIMIT core test set is shown in table II. For these experiments both ML and MMI trained HMMs were used as the generative model to obtain the features. As expected the use of MMI training with the standard HMM gave large gains on this task over the ML trained systems. However the CAUG model gave reductions in error rate for both ML and MMI trained systems. As expected the gains for the ML baseline system were larger than when using MMI training, though the absolute performance of the MMI-based CAUG system was about 0.8% absolute better than the equivalent ML-based system.

Despite good performance compared to standard HMMs, CAug models do not quite attain the performance of HCRFs [13]. This is believed to be due to three main factors: the fixed state segmentation from the base model, over-training (training error falls to 15.1% for MMI statistics) and lack of a language model (tests on MMI HMMs suggest that this may yield a gain of up to 0.5% absolute). To handle the over-training issue it may be interesting to use, for example, the large margin criterion in equation 14 rather than CML.

X. Conclusion

This paper has reviewed some of the acoustic models used in acoustic speech recognition, with particular emphasis on discriminative models. The current forms of discriminative training that have been used to improve the performance of standard HMM-based systems were described. These discriminatively trained HMMs are used in the majority of state-of-the-art LVCSR systems. A number of possible discriminative models that have been proposed in the literature have been discussed along with a more detailed description of the conditional augmented model.

Though many of the various forms of discriminative model described have achieved gains over baseline systems (some with discriminative training), it is not clear whether these gains will map to LVCSR tasks. The tasks on which the discriminative models have typically been applied have been

relatively simple compared to the state-of-the-art LVCSR tasks. Related to this, it is not clear whether the techniques used to improve generalisation)language model weakening, I-smoothing etc) will work for these discriminative models. This may be particularly important given the additional flexibility of the discriminative models compared to the generative models.

ACKNOWLEDGEMENT

The author would like to thank Martin Layton for the many discussions about augmented statistical models.

REFERENCES

[1] M.J.F. Gales, D.Y. Kim, P.C. Woodland, H.Y. Chan, D. Mrva, R. Sinha, and S.E. Tranter, "Progress in the CU-HTK Broadcast News transcription system," *IEEE Transactions Audio, Speech and Language Processing*, 2006.

[2] C.J. Leggetter and P.C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density HMMs," *Computer Speech and Language*, vol. 9, pp. 171–186, 1995.

[3] T. Anastasakos, J. McDonough, R. Schwartz, and J. Makhoul, "A compact model for speaker-adaptive training," in *Proceedings ICSLP*, 1996, pp. 1137–1140.

[4] M.J.F. Gales, "Semi-tied covariance matrices for hidden Markov models," *IEEE Transactions on Speech and Audio Processing*, vol. 7, no. 3, pp. 272–281, 1999.

[5] L.A. Rabiner, "A tutorial on hidden Markov models and selective applications in speech recognition," in *Proc. of the IEEE*, February 1989, vol. 77, pp. 257–286.

[6] P.S. Gopalakrishnan, D. Kanevsky, A. Nádas, and D. Nahamoo, "An inequality for rational functions with applications to some statistical estimation problems," *IEEE Trans. Information Theory*, 1991.

[7] P. C. Woodland and D. Povey, "Large scale discriminative training of hidden Markov models for speech recognition," *Computer Speech & Language*, vol. 16, pp. 25–47, 2002.

[8] D. Povey and P. C. Woodland, "Minimum phone error and I-smoothing for improved discriminative training," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Orlando, FL, May 2002.

[9] D. Povey, *Discriminative Training for Large Vocabulary Speech Recognition*, Ph.D. thesis, Cambridge University, 2004.

[10] R. Schlüter, B. Müller, F. Wessel, and H. Ney, "Interdependence of language models and discriminative training," in *Proc. ASRU*, 1999.

[11] V. Valtchev, J. J. Odell, P. C. Woodland, and S. J. Young, "Lattice-based discriminative training for large vocabulary speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 1996.

[12] H-K. Kuo and Y. Gao, "Maximum entropy direct models for speech recognition," *IEEE Transactions Audio Speech and Language Processing*, 2006.

[13] A. Gunawardana, M. Mahajan, A. Acero, and J.C. Platt, "Hidden conditional random fields for phone classification," in *Interspeech*, 2005.

[14] M.I. Layton and M.J.F. Gales, "Augmented statistical models for speech recognition," in *ICASSP*, 2006.

[15] N.D. Smith and M.J.F. Gales, "Speech recognition using SVMs," in *Advances in Neural Information Processing Systems*, 2001.

[16] A.P. Dempster, N.M. Laird, and D.B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society*, vol. 39, pp. 1–39, 1977.

[17] P. Brown, *The Acoustic-Modelling Problem in Automatic Speech Recognition*, Ph.D. thesis, IBM T.J. Watson Research Center, 1987.

[18] B. Roark, M. Saraclar, and M. Collins, "Discriminative N-gram language modeling," *Computer Speech and Language*, 2007.

[19] B.-H. Juang and S. Katagiri, "Discriminative learning for minimum error classification," *IEEE Transactions on Signal Processing*, 1992.

[20] J. Kaiser, B. Horvat, and Z. Kacic, "A novel loss function for the overall risk criterion based discriminative training of HMM models," in *Proc. ICSLP*, 2000.

[21] W. Byrne, "Minimum Bayes risk estimation and decoding in large vocabulary continuous speech recognition," *IEICE Special Issue on Statistical Modelling for Speech Recognition*, 2006.

[22] W. Macherey, L. Haferkamp, R. Schlter, and H. Ney, "Investigations on error minimizing training criteria for discriminative training in automatic speech recognition," in *Proceedings Eurospeech*, 2005.

[23] G. Saon, D. Povey, and G. Zweig, "CTS decoding improvements at IBM," in *EARS STT workshop*, St. Thomas, U.S. Virgin Islands, December 2003.

[24] A. Stolcke, E. Brill, and M. Weintraub, "Explicit word error minimization in N-Best list rescoring," in *Proc. Eur. Conf. Speech Commun. Technol.*, 1997.

[25] L. Mangu, E. Brill, and A. Stolcke, "Finding consensus among words: Lattice-based word error minimization," in *Proc. Eurospeech*, 1999.

[26] V. Goel and W. Byrne, "Task dependent loss functions in speech recognition: A* search over recognition lattices," in *Proc. Eur. Conf. Speech Commun. Technol.*, 1999.

[27] M. Layton and M.J.F. Gales, "Maximum margin training of generative kernels," Tech. Rep. CUED/F-INFENG/TR.484, Department of Engineering, University of Cambridge, June 2004.

[28] H. Jiang, X. Li, and Liu X., "Large margin hidden markov models for speech recognition," *IEEE Transactions Audio, Speech and Language Processing*, September 2006.

[29] J. Li, M. Siniscalchi, and C-H. Lee, "Approximate test risk minimization theough soft margin training," in *ICASSP*, 2007.

[30] F. Sha and L.K. Saul, "Large margin gaussian mixture modelling for phonetic classification and recognition," in *ICASSP*, 2007.

[31] K. Papineni, "Discriminative training via linear programming," in *Proc. ICASSP*, 1999.

[32] B. Tasker, *Learning Structured Prediction Models: A Large Margin Approach*, Ph.D. thesis, Stanford University, 2004.

[33] J. Zheng and A. Stolke, "Improved discriminative training using phone lattices," in *Proceedings InterSpeech*, Lisbon, Portugal, September 2005.

[34] F. Sha and L.K. Saul, "Large margin gaussian mixture modelling for automatic speech recognition," in *Advances in Neural Information Processing Systems*, 2007.

[35] J. Lafferty, A. McCallum, and F. Pereira, "Condition random fields: Probabilistic models for segmenting and labeling sequence data," in *ICML*, 2001, pp. 591–598.

[36] J.A. Bilmes, "Buried Markov models: A graphical-modelling approach to automatic speech recognition," *Computer Speech and Language*, vol. 2-3, 2003.

[37] K. Tsuda, T. Kin, and K. Asai, "Marginalized kernels for biological sequences," *Bioinformatics*, vol. 18, pp. S268–S275, 2002.

[38] T. Jaakkola and D. Hausser, "Exploiting generative models in discriminative classifiers," in *Advances in Neural Information Processing Systems 11*, S.A. Solla and D.A. Cohn, Eds. 1999, pp. 487–493, MIT Press.

[39] C. Saunders, J. Shawe-Taylor, and A. Vinokourov, "String kernels, fisher kernels and finite state automata," in *Advances in Neural Information Processing Systems 15*, S. Becker, S. Thrun, and K. Obermayer, Eds. 2003, pp. 633–640, MIT Press.

[40] V.N. Vapnik, *Statistical learning theory*, John Wiley & Sons, 1998.

[41] V. Venkataramani, S. Chakrabartty, and W. Byrne, "Support vector machines for segmental minimum Bayes risk decoding of continuous speech," in *ASRU 2003*, 2003.

[42] M.J.F. Gales and M.I. Layton, "Training augmented models using svms," *IEICE Special Issue on Statistical Modelling for Speech Recognition*, 2006.

[43] V. Wan and S. Renals, "Speaker verification using sequence discriminant support vector machines," *IEEE Transactions Speech and Audio Processing*, 2004.

[44] C. Longworth and M.J.F. Gales, "Discriminative adaptation for speaker verification," in *Proceedings InterSpeech*, September 2006.

[45] N.D. Smith, *Using Augmented Statistical Models and Score Spaces for Classification*, Ph.D. thesis, University of Cambridge, September 2003.

[46] S.J. Young, G. Evermann, M.J.F. Gales, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P.C. Woodland, *The HTK Book, version 3.4*, Cambridge University Engineering Department, Cambridge, UK, 2006.

[47] M. Layton, *Augmented Statistical Moedls for Classifying Sequence Data*, Ph.D. thesis, Cambridge University, 2006.